

## **BAB III**

### **ANALISA DAN PERANCANGAN**

#### **3.1. Analisa Permasalahan**

Ayam merupakan kebutuhan manusia dalam memenuhi protein dalam tubuhnya. Ayam juga sebagai alternatif pengganti sapi dan kambing yang harganya lebih murah daripada harga sapi dan kambing. Disamping itu budidaya ternak ayam sangatlah mudah dan murah dalam pekerjaannya serta memiliki nilai keuntungan yang besar.

Ada beberapa aspek yang sangat rentang dalam ternak ayam yaitu kebutuhan nutrisi ayam dan ransumnya. Banyak peternak ayam yang kurang memperhatikan kebutuhan nutrisi sehingga memengaruhi manajemen keuangan sebuah peternak ayam. Oleh karena itu, agar kebutuhan nutrisi dapat terpenuhi diiringi dengan minimnya pengeluaran kebutuhan ransum maka dibuatlah sebuah aplikasi yang mana dapat mengetahui kandungan nutrisi, takaran, ransum dan harga yang bertujuan untuk dapat mengoptimalkan ternak ayam broiler.

#### **3.2. Analisa Kebutuhan**

Analisa kebutuhan dilakukan untuk mengetahui kebutuhan yang diperlukan untuk membangun aplikasi. Analisa dilakukan dengan mencari dan menentukan beberapa kebutuhan seperti mendeskripsikan sistem yang akan dibangun, fungsi-fungsi yang dibutuhkan, komponen-komponen apa saja yang akan terlibat dalam pembuatan aplikasi, data hasil proses sistem serta desain *interface*.

##### **3.2.1. Analisa Kebutuhan Fungsional**

Analisa kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan.

##### **3.2.2. Analisa Kebutuhan Non Fungsional**

Analisa kebutuhan non fungsional menggambarkan kebutuhan sistem yang menjelaskan mengenai perilaku yang dimiliki oleh sistem. Kebutuhan non fungsional aplikasi optimasi penentuan bahan pakan ayam broiler dengan menggunakan Metode seleksi *roulette wheel* adalah:

### 1. Kebutuhan Perangkat Lunak

Terdapat beberapa perangkat lunak yang digunakan dalam membangun aplikasi ini dengan memiliki spesifikasi sebagai berikut:

- a. Java Development Kit (JDK) 8
- b. Netbeans IDE 8.0
- c. Java library jexcelapi 2.6.12 dan poi 3.16
- d. Microsoft Excel 2010

### 2. Kebutuhan Perangkat Keras

Perangkat keras yang digunakan selama membangun aplikasi ini memiliki spesifikasi sebagai berikut:

- a. Processor : Intel Celeron CPU 1007U @1.50GHz
- b. Memory : 2.00 GB
- c. Storage : 500GB HDD
- d. Graphics : Intel HD Graphics

## 3.3. Deskripsi Umum Sistem

Sistem yang akan dibangun dalam penelitian ini adalah sebuah sistem yang mengimplementasikan Algoritma Genetika dalam permasalahan optimasi komposisi pakan ternak ayam broiler. Untuk mendapatkan kombinasi bahan pakan ternak, tersedia 50 bahan pakan ayam broiler. Pengguna akan memilih beberapa bahan pakan yang akan dijadikan kombinasi pakan ayam broiler.

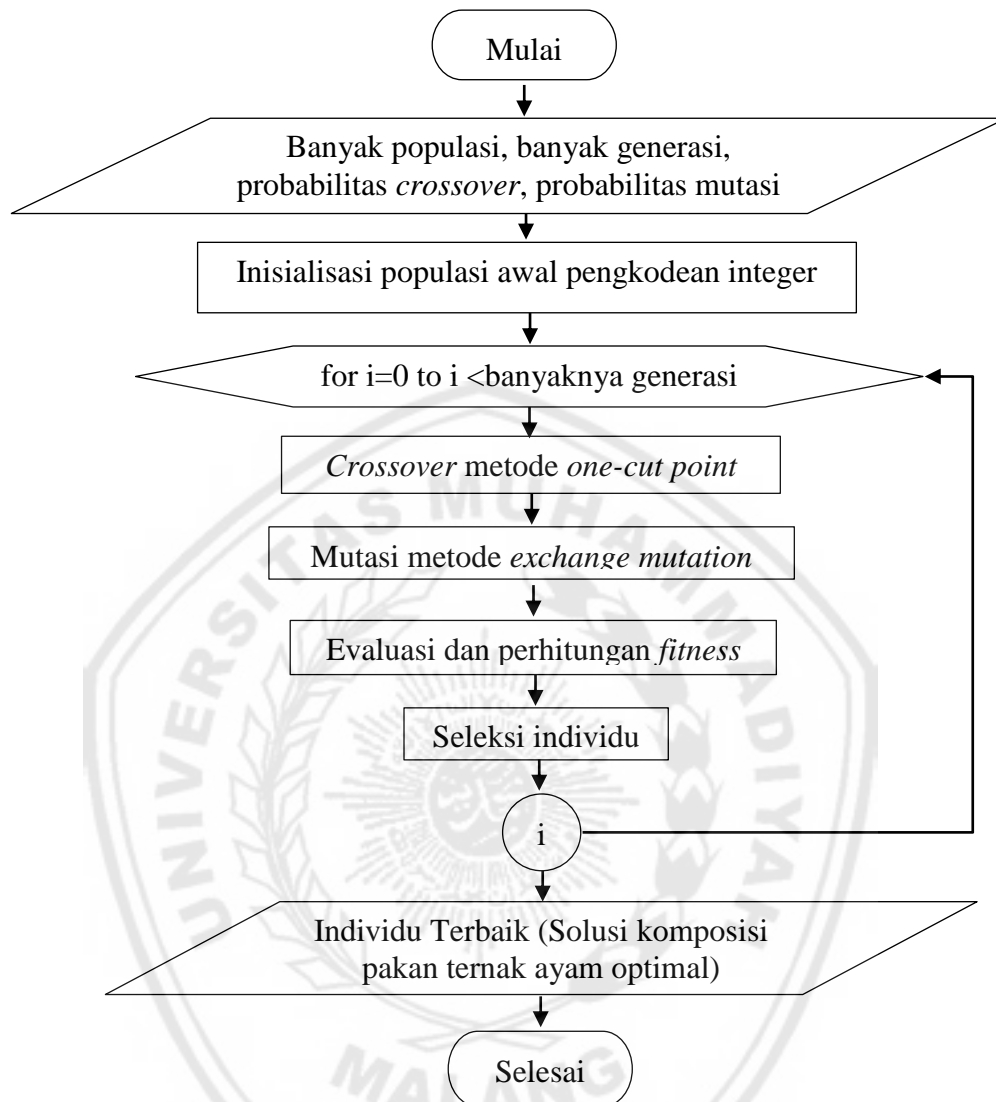
Selain itu terdapat *input*-an yaitu kandungan nutrisi, kebutuhan nutrisi dan harga bahan pakan ayam broiler, jenis fase ayam broiler (*starter* dan *finisher*), banyak ayam, probabilitas *crossover* (pc), probabilitas mutasi (pm), serta jumlah iterasi yang diinginkan *user*. Algoritma Genetika diharapkan dapat menentukan solusi dalam optimasi komposisi pakan ternak ayam broiler dengan komposisi yang tepat yaitu terpenuhinya kebutuhan nutrisi harian ayam broiler namun dengan biaya pembelian pakan yang minimal.

## 3.4. Proses Algoritma Genetika

Proses optimasi komposisi pakan ternak ayam broiler dengan menggunakan Algoritma Genetika adalah sebagai berikut :

1. Representasi kromosom/ inisialisasi parameter awal, yaitu:
  - a) Parameter optimasi pakan ternak ayam broiler yaitu fase ayam broiler yang terdiri dari *starter* (0 - 3 minggu) dan *finisher* (3 - 6 minggu).
  - b) Parameter algoritma genetika, antara lain:
    - Jumlah populasi (*popsize*)
    - Banyak bahan pakan yang dipilih
    - Probabilitas *crossover* (*Pc*)
    - Probabilitas mutasi (*Pm*)
    - Jumlah iterasi
  - c) Nilai **Pc** berkisar antara 0 dan 0.9 begitupula dengan nilai **Pm** antara 0 dan 0.9. Hal ini berdasarkan banyak studi kasus pada jurnal yang memakai nilai *Pc* dan *Pm* antara 0 hingga 0.9 yang mana dengan kombinasi nilai tersebut sudah membuktikan bahwa nilai *fitness* menjadi nilai *fitness* terbaik.
2. Inisialisasi populasi awal sebanyak jumlah populasi yang telah ditentukan. Panjang kromosom dari individu sesuai dengan banyaknya bahan pakan yang dipilih.
3. Reproduksi atau membuat populasi baru dengan langkah sebagai berikut:
  - Melakukan proses *crossover* pada induk yang terpilih secara random berdasarkan *Pc* yang telah ditentukan.
  - Melakukan proses mutasi pada induk yang terpilih secara random berdasarkan *Pm* yang telah ditentukan.
  - Menghitung nilai *fitness* untuk masing-masing individu.
  - Seleksi dengan menggunakan metode *Roulette wheel* untuk menentukan individu pada proses berikutnya.
  - Proses populasi baru yaitu dengan memilih individu sebanyak jumlah populasi awal setelah dilakukannya proses seleksi yang akan menjadi populasi baru untuk generasi berikutnya.
4. Jika kondisi akhir terpenuhi yaitu yaitu pada generasi yang ditentukan, maka kondisi berhenti dan hasilnya adalah solusi terbaik dari individu saat itu

yang memiliki nilai *fitness* terbesar pada seluruh generasi. *Flowchart* untuk proses Algoritma Genetika yang digunakan dapat dilihat pada Gambar 3.1



**Gambar 3.1. Flowchart Proses AG**

Langkah-langkah Algoritma Genetika yang diimplimentasikan dapat dijelaskan pada subbab berikut :

#### **3.4.1. Inisialisasi Populasi Awal**

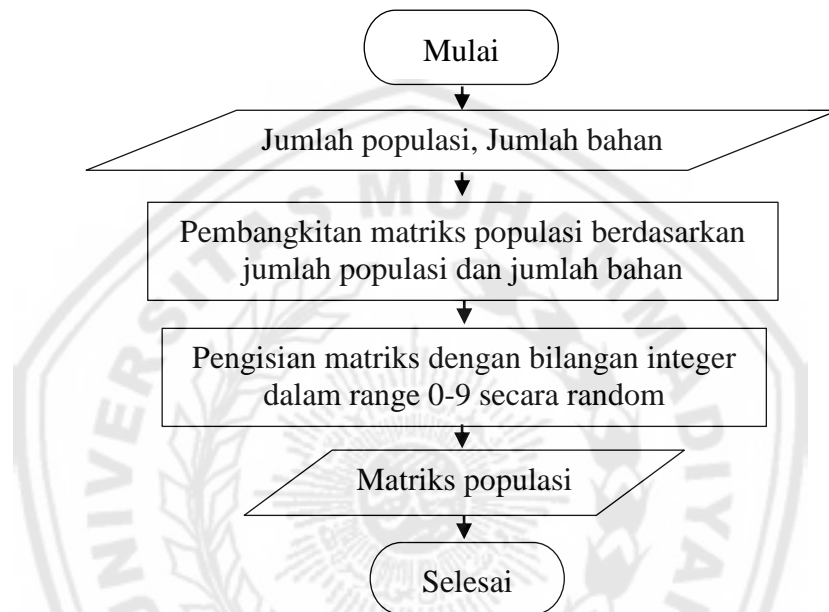
Proses inisialisasi merupakan proses pembangkitan kromosom sebanyak jumlah populasi awal yang telah ditentukan. Berikut langkah-langkah pembangkitan populasi awal :

1. Menentukan jumlah populasi.

Lakukan proses inisialisasi kromosom sebanyak populasi awal.

- a. Menentukan panjang kromosom individu sesuai dengan komposisi bahan pakan yang dipilih dari 50 bahan pakan yang telah ditentukan/tersedia.
- b. Mengisi kromosom dengan membangkitkan bilangan integer secara acak antara 0-9. Setiap kromosom mewakili bahan pakan yang dipilih oleh *user*.

*Flowchart* untuk proses inisialisasi yang digunakan dapat dilihat pada Gambar 3.2.



**Gambar 3.2. Flowchart Inisialisasi**

#### 3.4.2. Pindah Silang (Crossover)

Setelah proses pembangkitan populasi awal, proses selanjutnya yaitu reproduksi yang terdiri dari *crossover* dan mutasi. Individu-individu dari hasil pembangkitan populasi akan dipilih secara acak sebagai induk untuk dilakukan proses *crossover*. Dalam tahap ini harus ditentukan tingkat *crossover rate* ( $P_c$ ) untuk menyatakan banyaknya *offspring* yang dihasilkan. Pada laporan skripsi ini proses *crossover* menggunakan metode *one-cut point* dengan langkah-langkah berikut :

1. Menghitung jumlah *offspring* yang akan dihasilkan yaitu tergantung dari besarnya  $P_c$  dikalikan jumlah populasi (*popsiz*e).

2. Memilih dua induk secara acak tergantung dengan banyaknya *offspring*. Karena setiap satu kali proses *crossover* menghasilkan dua anak (*child*) dan satu kali proses *crossover* menyilangkan dua induk.
3. Memilih satu titik potong (*cut point*) secara acak dan menukarkan kromosom *parent* 1 bagian kanan *cut point* dengan kromosom *parent* 2 bagian kanan *cut point* untuk menghasilkan *offspring*

#### **3.4.3. Mutasi**

Proses reproduksi selanjutnya yaitu proses mutasi. Peluang individu mengalami mutasi berdasarkan nilai  $P_m$  yang telah ditentukan terhadap jumlah keseluruhan gen hasil *generate* populasi. Pada laporan skripsi ini proses mutasi menggunakan metode *reciprocal exchange mutation* dengan langkah-langkah berikut :

1. Menghitung jumlah *offspring* yang akan dihasilkan yaitu tergantung dari besarnya  $P_m$  dikalikan jumlah populasi (*popsize*).
2. Memilih satu induk secara acak tergantung dengan banyaknya *offspring*. Karena setiap satu kali proses mutasi menghasilkan satu anak (*child*).
3. Memilih dua indeks untuk dijadikan *mutation point* (*exchange point* / *XP*) secara random kemudian menukarkan nilai pada indeks yang terpilih sebagai *mutation point*.

#### **3.4.4. Perhitungan *Fitness***

Setelah didapatkan *child* hasil dari *crossover* dan mutasi, selanjutnya adalah melakukan perhitungan *fitness* sesuai dengan (Persamaan 2.3) untuk semua individu. Hasil perhitungan *fitness* kemudian akan dijadikan masukan proses seleksi.

#### **3.4.5. Seleksi**

Proses seleksi dilakukan dengan menggunakan metode *Roulette Wheel*. Proses ini dilakukan dengan tujuan untuk memperoleh individu terpilih untuk dijadikan populasi baru yang akan digunakan untuk proses evaluasi berikutnya sebanyak jumlah populasi.

### 3.5. Penerapan Metode Algoritma Genetika

#### 3.5.1. Inisialisasi kromosom

Inisialisasi kromosom dibangkitkan sebanyak *PopSize* dan satu bahan pakan dianggap sebagai satu gen penyusun individu. Jika terdapat lima bahan pakan, maka panjang kromosom adalah lima. Pembangkitan nilai kromosom dilakukan dengan membangkitkan nilai integer pada interval antara 0-9 secara *random*.

Misalkan inisialisasi kromosom dari 5 bahan pakan yang dipilih yaitu *biji sorgum*, *dedak jagung halus*, *jagung kuning*, *konsentrat layer* dan *tepung bulu* dengan popsize = 10, pc = 0.2, pm = 0.1 dan iterasi = 1 dapat dilihat pada Tabel 3.1 setiap satu kromosom mewakili bahan pakan.

**Tabel 3.1. Inisialisasi Kromosom**

Individu	Kromosom				
P1	7	1	8	7	3
P2	6	2	0	8	1
P3	8	7	8	2	1
P4	7	4	1	4	1
P5	5	2	0	2	7
P6	6	8	7	0	9
P7	0	7	6	4	3
P8	0	2	2	4	1
P9	0	7	4	5	4
P10	1	5	6	2	3

#### 3.5.2. Reproduksi

Pada tahap ini terdapat dua operator genetika yaitu proses *crossover* dan mutasi. Metode *crossover* yang digunakan adalah *one-cut point* yang menghasilkan *offspring* dari kombinasi nilai dua induk. Dalam tahap ini harus ditentukan probabilitas *crossover* (Pc) yang menyatakan rasio *offspring* yang dihasilkan proses *crossover* terhadap ukuran populasi sehingga akan dihasilkan

*offspring* sebanyak  $P_c \times PopSize$ . Misalkan diketahui bahwa nilai  $P_c$  adalah 0.2 dan  $PopSize$  10, maka *offspring* yang dihasilkan adalah  $0.2 \times 10 = 2$ . Selanjutnya memilih dua individu secara random untuk dilakukan proses *crossover*, misalnya P2 dan P3 adalah individu yang terpilih, maka *offspring* 1 dan *offspring* 2 bisa dibangkitkan.

Sesuai pada Tabel 3.1 nilai pada individu P2 dan P3 adalah P2 = 6 2 0 8 1 dan P3 = 8 7 8 2 1. Kemudian memilih *cut point* secara *random* untuk dilakukan proses penukaran kromosom dua individu yang terpilih, misalkan *cut point* yang terpilih pada indeks 0. Maka *offspring* yang dihasilkan seperti berikut.

P2 = 6 2 0 8 1

P3 = 8 7 8 2 1

*Offspring* 1 = 6 7 8 2 1 dan *Offspring* 2 = 8 2 0 8 1

Metode mutasi yang digunakan adalah *exchange* yang menghasilkan *offspring* dari satu induk. Pada tahap ini harus ditentukan probabilitas mutasi ( $P_m$ ) yang menyatakan rasio *offspring* yang dihasilkan proses mutasi terhadap ukuran populasi sehingga akan dihasilkan *offspring* sebanyak  $P_m \times PopSize$ . Misal diketahui bahwa nilai  $P_m$  adalah 0.1 dan  $PopSize$  10, maka *offspring* yang dihasilkan adalah  $0.1 \times 10 = 1$ . Selanjutnya memilih satu individu secara *random* untuk dilakukan proses mutasi, misalnya P10 adalah individu yang terpilih, maka *offspring* bisa dibangkitkan.

Sesuai pada Tabel 3.1 nilai pada individu P10 adalah P10 = 1 5 6 2 3. Kemudian memilih dua *point* secara *random* untuk dilakukan proses penukaran kromosom, misalkan *point* yang terpilih pada indeks 0 dan 1. Maka *offspring* yang dihasilkan seperti berikut.

P10 = 1 5 6 2 3

*Offspring* 3 = 5 1 6 2 3

### 3.5.3 Evaluasi dan perhitungan *fitness*

**Tabel 3.2. Evaluasi**

Individu	Kromosom					Total
P1	7	1	8	7	3	26
P2	6	2	0	8	1	17
P3	8	7	8	2	1	26



Individu	Kromosom					Total
P4	7	4	1	4	1	17
P5	5	2	0	2	7	16
P6	6	8	7	0	9	30
P7	0	7	6	4	3	20
P8	0	2	2	4	1	9
P9	0	7	4	5	4	20
P10	1	5	6	2	3	17
Offspring 1	6	7	8	2	1	24
Offspring 2	8	2	0	8	1	19
Offspring 3	5	1	6	2	3	17

Pada tahap ini inisialisasi kromosom dalam bentuk integer dirubah dalam komposisi gr dengan rumus:

$$P_i = \frac{\text{Kromosom } [i]}{\text{Total}} \times \text{kemampuan konsumsi ayam per hari} \quad (3.1)$$

Sesuai dengan Persamaan 3.1, maka komposisi dalam bentuk gr didapatkan sebagai berikut.

P1 ->  $\frac{7}{26} \times 44 = 11.8462$  gr untuk biji sorgum,  $\frac{1}{26} \times 44 = 1.6923$  gr untuk dedak jagung halus,  $\frac{8}{26} \times 44 = 13.5385$  gr untuk jagung kuning,  $\frac{7}{26} \times 44$  gr untuk konsentrat layer = 11.8462 dan  $\frac{3}{26} \times 44 = 5.0769$  gr untuk tepung bulu. Hasil dari komposisi pakan dalam bentuk gram ditunjukkan pada Tabel 3.3

**Tabel 3.3. Komposisi Dalam Bentuk gr**

Individu	Komposisi (gr)					Total
P1	11.8462	1.6922	13.5385	11.8462	5.0769	44

$$P_i = \text{Kromosom } [i] \times \text{harga pakan} \quad (3.2)$$

Sesuai dengan Persamaan 3.2, maka harga komposisi dalam bentuk gr didapatkan sebagai berikut.

Diketahui harga bahan pakan yaitu biji sorgum Rp 2500/kg, dedak jagung halus 2000/kg, jagung kuning Rp 3000/kg, konsentrat layer Rp 2500/kg dan tepung bulu Rp 4000/kg, maka P1 adalah:

$$11.8462 \times 2500 = 29615.5,$$

$$1.6922 \times 2000 = 3384.4,$$

$$13.5385 \times 3000 = 40615.5,$$

$$11.8462 \times 2500 = 29615.5,$$

$$5.0769 \times 4000 = 20307.6.$$

Jadi total harga untuk P1 adalah

$$7095 + 25548.4 + 8514 + 31935.5 + 51096.8 = 124189.7$$

Hasil dari harga pakan ditunjukkan pada Tabel 3.4.

**Tabel 3.4. Harga Komposisi Bahan Pakan**

Individu	Harga					Total
P1	29615.5	3384.4	40615.5	29615.5	20307.6	123538.5

Menghitung EM, PK, SK, Ca dan P untuk digunakan sebagai perhitungan *fitness*. Sebelum menghitung pemenuhan nutrisi dilakukan perhitungan kandungan nutrisi masing-masing bahan pakan sesuai dengan rumus:

$$KN = \text{Kromosom}[i] \times NBp \quad (3.3)$$

Keterangan :

KN = kandungan nutrisi

Kromosom [i] = kromosom dalam komposisi gram

NBp = kandungan nutrisi bahan pakan

Kandungan nutrisi pada biji sorgum, dedak jagung kuning, jagung kuning, konsentrat layer dan tepung bulu yaitu:

**Tabel 3.5 Kandungan Nutrisi**

BAHAN PAKAN	EM	PK	SK	Ca	P
	Kkal/kg	%	%	%	%
Biji sorgum	3250	10	2	0.41	2.13
Dedak jagung halus	2950	7.54	0.58	0.51	0.15
Jagung Kuning	3370	8.6	2	0.09	0.37
Konsetrat Layer	2500	32	6	3.4	2.1
Tepung bulu	2310	85	1.5	0.32	0.32

Sehingga untuk perhitungan P1 adalah:

Biji sorgum :  $Em = 11.8462 \times 3250 = 38500.2$

$$PK = 11.8462 \times \frac{10}{100} = 1.1846$$

$$SK = 11.8462 \times \frac{2}{100} = 0.2369$$

$$Ca = 11.8462 \times \frac{0.41}{100} = 0.0486$$

$$P = 11.8462 \times \frac{2.13}{100} = 0.2523$$

Dedak jagung halus :  $Em = 1.6922 \times 2950 = 4992$

$$PK = 1.6922 \times \frac{7.54}{100} = 0.1276$$

$$SK = 1.6922 \times \frac{0.58}{100} = 0.0098$$

$$Ca = 1.6922 \times \frac{0.51}{100} = 0.0086$$

$$P = 1.6922 \times \frac{0.15}{100} = 0.0025$$

Jagung kuning :  $Em = 13.5385 \times 3370 = 45624.7$

$$PK = 13.5385 \times \frac{8.6}{100} = 1.1643$$

$$SK = 13.5385 \times \frac{2}{100} = 0.2708$$

$$Ca = 13.5385 \times \frac{0.09}{100} = 0.0122$$

$$P = 13.5385 \times \frac{0.37}{100} = 0.0501$$

Konsetrat layer :  $Em = 11.8462 \times 2500 = 28715.5$

$$PK = 11.8462 \times \frac{32}{100} = 3.7908$$

$$SK = 11.8462 \times \frac{6}{100} = 0.7108$$

$$Ca = 11.8462 \times \frac{3.4}{100} = 0.4028$$

$$P = 11.8462 \times \frac{2.1}{100} = 0.2488$$

Tepung bulu :  $Em = 5.0769 \times 2310 = 11727.6$

$$PK = 5.0769 \times \frac{85}{100} = 4.3154$$

$$SK = 5.0769 \times \frac{1.5}{100} = 0.0762$$

$$Ca = 5.0769 \times \frac{0.32}{100} = 0.0162$$

$$P = 5.0769 \times \frac{0.32}{100} = 0.0162$$

Setelah didapatkan kandungan nutrisi tiap bahan pakan, selanjutnya menjumlah masing-masing nutrisi. Berikut total dari kandungan nutrisi :

1. EM =  $38500.2 + 4992 + 45624.7 + 28715.5 + 11727.6 = 129560$
2. PK =  $1.1846 + 0.1276 + 1.1643 + 3.7908 + 4.3154 = 10.5827$
3. SK =  $0.2369 + 0.0098 + 0.2708 + 0.7108 + 0.0762 = 1.3045$
4. Ca =  $0.0486 + 0.0086 + 0.0122 + 0.4028 + 0.0162 = 0.4884$
5. P =  $0.2523 + 0.0025 + 0.0501 + 0.2488 + 0.0162 = 0.5699$

Kemudian apabila total kandungan nutrisi lebih besar dari kebutuhan nutrisi, maka nutrisi tersebut terpenuhi sedangkan apabila total kandungan nutrisi lebih kecil dari kebutuhan nutrisi, maka menghitung kekurangan nutrisi sesuai dengan Persamaan 3.4.

$$\text{KurangNutrisi} = \text{kebutuhan nutrisi} - \text{total kandunganNutrisi} \quad (3.4)$$

Maka pemenuhan nutrisi untuk P1 adalah

1. EM = 132000 kkal yang didapat = 129560 kkal (tidak terpenuhi)  
kekurangan = 2440
2. PK = 10.12 gr yang didapat = 10.5827 gr (terpenuhi)
3. SK = 2.2 gr yang didapat = 1.3045 gr (tidak terpenuhi) kekurangan = 0.8955
4. Ca = 0.44 gr yang didapat = 0.4884 gr (terpenuhi)
5. P = 0.22 gr yang didapat = 0.5699 gr (terpenuhi)

Jadi total kekurangan nutrisi dari P1 adalah  $2440 + 0.8955 = 2440.8955$ .

Hasil dari perhitungan pemenuhan nutrisi dari semua individu ditunjukkan pada Tabel 3.6.

**Tabel 3.6. Kekurangan Nutrisi**

Individu	Total kekurangan nutrisi
P1	2440.8955

Setelah harga dan kekurangan nutrisi diketahui, selanjutnya melakukan perhitungan nilai *fitness* pada semua individu. Perhitungan nilai *fitness* menggunakan rumus *fitness* pada Persamaan 3.5.

$$\text{Fitness} = \frac{100}{(\text{harga} + (\text{kekurangan nutrisi} * K))} \quad (3.5)$$

Keterangan :

- Harga : Total harga dari semua bahan pakan
- Kekurangan Nutrisi : Nilai nutrisi yang kurang dari yang dibutuhkan
- K : Nilai konstanta pengali untuk kekurangan nutrisi

Pada kasus komposisi pakan ternak ayam broiler ini, nilai konstanta yang digunakan adalah 100. Nilai 100 merupakan rata-rata total harga bahan pakan dari solusi yang diberikan. Hal ini dilakukan dengan tujuan agar nilai kekurangan nutrisi dapat memiliki nilai yang sebanding dengan nilai harga. Pada akhirnya, harga dan kekurangan nutrisi memiliki tingkat kepentingan yang sama dengan tujuan agar individu yang terpilih sebagai solusi merupakan individu yang fokus pada proses meminimalkan nilai kekurangan nutrisi dan harga biaya pakan ternak ayam broiler. Sehingga didapatkan nilai *fitness* sebagai berikut.

$$Fitness \text{ untuk P1} = \frac{100}{\frac{120000}{1000} + (2440.8955 * 100)} = 0.000409, \text{ harga dibulatkan dan dibagi}$$

1000 karena komposisi berupa gr.

Hasil dari perhitungan nilai *fitness* dari semua individu ialah

**Tabel 3.7. Nilai *fitness***

Individu	Total Harga	Kurang Nutrisi	<i>Fitness</i>
P1	120000	2440.8955	0.000409
P2	110000	8515.7782	0.000117
P3	110000	5.7397	0.146205
P4	110000	1996.3579	0.000500
P5	140000	12871.3812	0.000078
P6	130000	3697.8453	0.000270
P7	120000	4841.249	0.000207
P8	120000	10022.6011	0.000100
P9	120000	9087.1427	0.000110
P10	120000	2201.384	0.000454
<i>Offspring 1</i>	110000	5.61	0.149031
<i>Offspring 2</i>	110000	6461.6763	0.000155
<i>Offspring 3</i>	130000	1.3474	0.377729

### 3.5.4. Seleksi

Memutar *Roulette wheel* untuk memilih setiap individu dengan langkah-langkah berikut:

- 1) Membangkitkan nilai random sebanyak *popsiz*.
- 2) Memilih / cek nilai random mulai dari 1 sampai *popsiz*, apabila nilai  $probCum_k$  dari suatu individu lebih besar dari nilai random, maka individu tersebut yang terpilih sebagai populasi baru
- 3) Setelah didapatkan individu populasi baru, selanjutnya adalah menentukan individu terbaik sebanyak jumlah populasi awal. Dengan mekanisme *Roulette wheel* maka individu dengan *fitness* lebih besar akan mempunyai peluang lebih besar untuk terpilih. Karena berupa peluang maka tidak menjamin bahwa individu yang memiliki *fitness* lebih besar akan selalu terpilih untuk masuk pada generasi berikutnya.

**Tabel 3.8. Nilai Probabilitas dan Probabilitas Komulatif**

Individu	Total Harga	Kurang Nutrisi	<i>Fitness</i>	Probabilitas	ProbCum
P1	120	2440.8955	0.000409	0.0006	0.0006
P2	110	8515.7782	0.000117	0.0002	0.0008
P3	110	5.7397	0.146205	0.2165	0.2173
P4	110	1996.3579	0.000500	0.0007	0.2180
P5	140	12871.3812	0.000078	0.0001	0.2181
P6	130	3697.8453	0.000270	0.0004	0.2185
P7	120	4841.249	0.000207	0.0003	0.2188
P8	120	10022.6011	0.000100	0.0001	0.2189
P9	120	9087.1427	0.000110	0.0002	0.2191
P10	120	2201.384	0.000454	0.0007	0.2198
Offspring 1	110	5.61	0.149031	0.2207	0.4405
Offspring 2	110	6461.6763	0.000155	0.0002	0.4407
Offspring 3	130	1.3474	0.377729	0.5593	1.0000

**Tabel 3.9. Hasil Roulette Wheel**

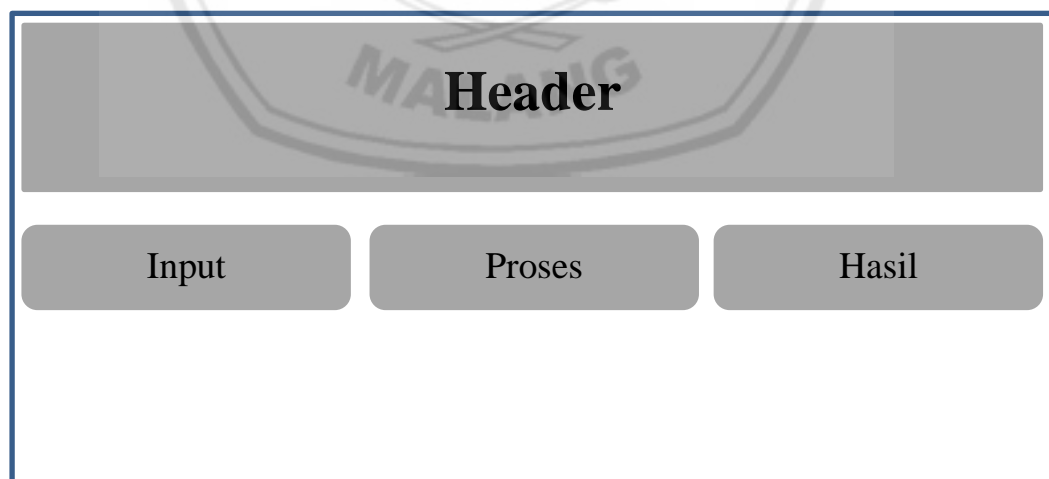
Individu	Random	Individu Terpilih	Kromosom					Fitness
P1	0,7601	P13	5	1	6	2	3	0.377729
P2	0,3670	P11	6	7	8	2	1	0.149031
P3	0,4213	P11	6	7	8	2	1	0.149031
P4	0,6705	P13	5	1	6	2	3	0.377729
P5	0,0162	P3	8	7	8	2	1	0.146205
P6	0,5060	P13	5	1	6	2	3	0.377729
P7	0,2167	P3	8	7	8	2	1	0.146205
P8	0,8993	P13	5	1	6	2	3	0.377729
P9	0,1215	P3	8	7	8	2	1	0.146205
P10	0,1934	P11	6	7	8	2	1	0.149031

Individu terbaik berdasarkan nilai *fitness* tertinggi. Hasil kromosom terbaik pada generasi pertama ialah

**Tabel 3.10. Hasil Individu Terbaik**

Individu	Komposisi (gr)					Total (gr)
P13 ( <i>offspring 3</i> )	5	1	6	2	3	17

### 3.6. Perancangan Interface



**Gambar 3.3. Tampilan Menu Utama**

Pada Gambar 3.3 merupakan tampilan awal dari aplikasi komposisi pakan ternak dimana pada tampilan awal ini terdiri dari 3 menu yaitu input, proses dan hasil.

**Header**

**Input**

Popsize

Pc

Pm

Iterasi

Proses

Bahan Pakan

**Gambar 3.4. Tampilan Pada Menu Input**

Pada Gambar 3.4 terdapat beberapa inisialisasi parameter awal yang mana akan diisi kemudian diproses dan dapat dilihat hasil dari pengkodean tersebut pada menu berikutnya.